## search & replace

Basic search and replace
The :subst itute command searches for a text pattern, and replaces it with a text string. There are many options, but these are what you probably want:
:%s/fo o/bar/g
Find each occurrence of 'foo' (in all lines), and replace it with 'bar'.
:s/foo /bar/g
Find each occurrence of 'foo' (in the current line only), and replace it with 'bar'.
:%s/fo o/b ar/gc
Change each 'foo' to 'bar', but ask for confir mation first.
:%s/\< foo \>/ bar/gc
Change only whole words exactly matching 'foo' to 'bar'; ask for confir mation.
:%s/fo o/b ar/gci
Change each 'foo' (case insens - itive due to the i flag) to 'bar'; ask for confir mation.
:%s/fo o\c /bar/gc is the same because \c makes the search case insens itive.
This may be wanted after using :set noigno recase to make searches case sensitive (the default).
:%s/fo o/b ar/gcI
Change each 'foo' (case sensitive due to the I flag) to 'bar'; ask for confir mation.
:%s/fo o\C /bar/gc is the same because \C makes the search case sensitive.

## search & replace (cont)

> This may be wanted after using :set ignorecase to make searches case insens- itive.
Search range:
:s/foo/bar/g Change each 'foo' to 'bar' in the current line.
:%s/foo/bar/g Change each 'foo' to 'bar' in all the lines.
:5,12s/foo/bar/g Change each 'foo' to 'bar' for all lines from line 5 to line 12 (inclusive).
:'a,'bs/foo/bar/g Change each 'foo' to 'bar' for all lines from mark a to mark b inclusive (see Note below).
:'<,'>s/foo/bar/g When compiled with +visual, change each 'foo' to 'bar' for all lines within a visual selection. Vim automa- tically appends the visual selection range ('<,'>) for any ex command when you select an area and enter :. Also, see Note below.
:.,$s/foo/bar/g Change each 'foo' to 'bar' for all lines from the current line (.) to the last line ($) inclusive.
:.,+2s/foo/bar/g Change each 'foo' to 'bar' for the current line (.) and the two next lines (+2).
:g/^baz/s/foo/bar/g Change each 'foo' to 'bar' in each line starting with 'baz'.

## search & replace (cont)

> Note: As of Vim 7.3, substitutions applied to a range defined by marks or a visual selection (which uses a special type of marks '< and '>) are not bounded by the column position of the marks by default. Instead, Vim applies the substitution to the entire line on which each mark appears unless the \%V atom is used in the pattern like: :'<,'>s/\%Vfoo/bar/g.

## Delete

Cancellare
x X cancella carattere al, prima del cursore
dm cancella il testo del comando di movimento m
dd D cancella la riga, fino alla fine della riga corrente
J gJ unisci la riga corrente succes siva, senza spazi
:rd↵ cancella il gruppo di righe r
:rdx↵ sposta il gruppo di righe r nel registro x

## copying

"x usa registro x per la prossima cancellazione, copia
:reg↵ mostra il contenuto di tutti i registri
:reg x↵ mostra il contenuto del registro x
ym copia il testo del comando di movimento m
yy or Y copia la riga corrente nel registro
p P metti registro dopo, prima del cursore
]p [p come p, P con indent - azione modificata

By **Max** (maxbini)
cheatography.com/maxbini/

Published 21st May, 2015.
Last updated 11th January, 2019.


Sponsored by **ApolloPad.com**
Everyone has a novel in them. Finish Yours!
https://apollopad.com

## copying (cont)

> gp gP come p, P lasciando il cursore dopo nuovo testo

## editing

**refresh file**

:edit

This means replace the end of line repres ented by $ with the string |.

:%s/$/|/

delete blank lines

:%s,\n \n,^M,g

:g/^$/d

:v/./d

delete all lines that are empty or that contain only whitespace characters

:g/^\s*$/d

:v/\S/d``

## grepping & sorting

grep

:vimgrep

sort string

:{rang e}sort u

sort number

:sort n

sort after " |"

:sort /^.*|/

## cut & paste

**REGISTERS**

**Normal registers**

a-z

**Clipboard**

+

**Numbered registers**

0 - last yanked text

1 - 9 last 9 deleted chunks of text

**Copy**

" <re gis ter >y< mot ion>

**Paste**

" <re gis ter >p

**SYSTEM CLIPBOARD**

**Copy**

" +y< mot ion>

**Paste**

"+p

## buffer

The :b command can also take a substring of the name of the file

:b bar will switch to the bar file.

:b o will switch to the foo file.

:b a will give you an error because it could mean either bar or baz, but you can fix the name to be more specific with, for example, : ↑ r Enter.

This command can also take a number, if you want to use that:

:b 5

## Insert Mode

" editing/moving within current insert (Really useful)

<C- U> : delete all entered

<C- W> : delete last word

<HO ME> <EN D> : beginn ing/end of line

<C- LEF TAR ROW ><C -RI GHT - ARR OW> : jump one word backwa - rds /fo rwards

<C- X>< C-E >,< C-X ><C -Y> : scroll while staying put in insert

--

Modalita' inseri mento

Vc Vn inserice car. c letterale, valore decimale di n

^A inserisce il testo preced - ent emente inserito

@ come A e va da modalita' inseri mento → comandi

Rx R^Rx inserisce contenuto del registro x, letterale

N P comple tamento del testo prima, dopo il cursore

^W cancella la parola prima del cursore

^U cancella tutti i caratteri inseriti nella riga corrente

D T sposta a sx, dx della larghezza di uno shift

^Kc1c2 or c1←c2 digita diagramma \c1,c2\

^Oc esegue c in modalita' comandi temporanea

XE XY scorri in su, giu'

&l t;e sc>\o ^[ vai modal. inseri mento → modal. comandi



Sponsored by **ApolloPad.com**
Everyone has a novel in them. Finish Yours!
https://apollopad.com

# Max VIM cheatsheet Cheat Sheet

by Max (maxbini) via cheatography.com/21458/cs/4138/

## duplicates

```
hilight duplicates
:syn clear Repeat | g/\(.\)\n\z e\%(.\n\)\1$/exe 'syn match Repeat "' .
escape (ge tli ne( '.'),
'".^$[]') . '$"' | nohlsearch
```

## moving

```
Movimenti base
h l k j un carattere a sx, dx,
su, giu'
b w parola /token a sx, dx
ge e fine parola /token sx, dx
{ } inizio preced ent e/p -
rossimo paragrafo
( ) inizio preced ente, prossima
frase
0 gm inizio, meta' riga
^ $ primo, ultimo carattere
della riga
nG ngg riga n, default ultima,
prima
n% n percento del file (n deve
essere indicato)
n| la colonna n della riga
corrente
% trova prossima parentesi,
commento, #define
nH nL riga n dall'i nizio, fine
della finestra
M alla riga di mezzo della
finestra
```

## Folding

```
zf#j creates a fold from the
cursor down # lines.
zf/string creates a fold from
the cursor to string .
zj moves the cursor to the next
fold.
zk moves the cursor to the
previous fold.
zo opens a fold at the cursor.
zO opens all folds at the
cursor.
```

## Folding (cont)

```
> zm increases the foldlevel by one.
zM closes all open folds.
zr decreases the foldlevel by one.
zR decreases the foldlevel to zero -- all
folds will be open.
zd deletes the fold at the cursor.
zE deletes all folds.
[z move to start of open fold.
]z move to end of open fold.
```

## Changing Case

```
guu : lowercase line
gUU : uppercase line
Vu : lowercase line
VU : uppercase line
g~~ : flip case line
vEU : Upper Case Word
vE~ : Flip Case Word
ggguG : lowercase entire file
```

## CR+LF

```
VI: visualizzare e rimuovere
CR+LF (^M)
- per visual izz arli:
:e ++ff=unix
- per visual izzarli di default:
set ff=unix
set filefo rma ts=unix
- per rimuov erli:
```

## CR+LF (cont)

```
> :%s/^M$//g
Note that you should type ^M by pressing
ctrl-v and then ctrl-m.
```

## insert

```
Inserisci &amp;amp; rimpiazza →
modalita' inserimento
i a inserisci prima, dopo il
cursore
I A inserisci all'in izio, fine
della riga
gI inserisci il testo nella
prima colonna
o O inserisci nuova riga sotto,
sopra riga corrente
rc rimpiazza il carattere al
cursore con c
grc come r, ma vai in modalita'
inseri mento
R rimpiazza caratteri a partire
dal cursore
gR come R, ma vai in mod.
inseri mento
cm cambia il testo del comando
di movimento m
cc or S cambia la riga corrente
C cambia fino alla fine della
riga
s cambia un carattere e vai in
mod. inseri mento
~ scambia maiusc olo /mi nuscolo
ad avanza il cursore
g~m scambia maiusc olo /mi -
nuscolo per cmd. mov. m
gum gUm minusc olo /ma iuscolo
testo cmd. mov. m
&amp;lt;m &amp;gt;m sposta sx, dx testo
del comando di mov. m
n& lt; &lt; n& gt; &gt; sposta n
righe a sx, dx
```

By **Max** (maxbini)
cheatography.com/maxbini/

Published 21st May, 2015.
Last updated 11th January, 2019.


Sponsored by **ApolloPad.com**
Everyone has a novel in them. Finish Yours!
https://apollopad.com

## Insert (advanced)

Inserimento avanzato
g?m esegui codifica rot13 su comando movimento m
n^A ^X +n, -n del numero al cursore
gqm formatta riga del movimento m a larghezza fissa
:rce w↵ centra le righe nel gruppo r a larghezza w
:rle i↵ allinea a sx righe gruppo r con indent. i
:rri w↵ allinea a dx righe gruppo r a larghezza w
!mc↵ filtra righe cmd. movimento m con cmd. c
n!!c↵ filtra le righe n con il comando c
:r!c↵ filtra gruppo di righe r con il comando c

## Macroing

per registrare la macro "a":
qa
per eseguire la macro " a"
@a
per ripetere l'ultima macro eseguita:
@@

## advanced

| | |
|---|---|
| remove multiple spaces in a visual selection | release visual selection, then: :'<,'>-s/\%\V\s\+/ /g |



Sponsored by **ApolloPad.com**
Everyone has a novel in them. Finish Yours!
https://apollopad.com